

Searching PAJ

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 06-019675
 (43)Date of publication of application : 28.01.1994

G06F 3/153

(51)Int.Cl.

(21)Application number : 04-172650
 (22)Date of filing : 30.06.1992

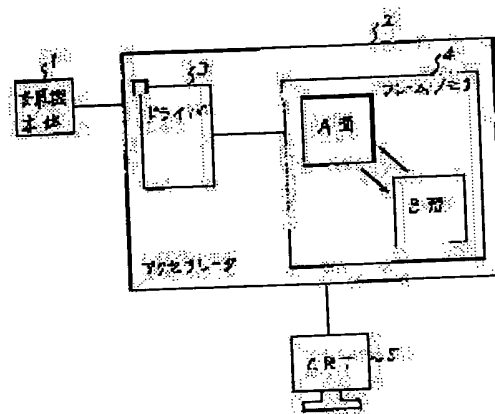
(71)Applicant : FUJITSU LTD
 (72)Inventor : KOBAYASHI TOMOHIRO

(54) GRAPHICS SYSTEM

(57)Abstract:

PURPOSE: To unnecessitate the complicated control considering a double-buffering function in an application program and to display the display contents to be outputted from the application program on a user screen at all times.

CONSTITUTION: When a side A is assumed as a display surface and a side B as a plotting surface, the content of the side A is displayed on a display 5. When a plotting request is sent from a computer main body 1, a driver 3 copies the whole contents of the display surface (the side A in this case) in the plotting surface (the side B in this case). At the end of copying, the picture specified by the plotting request is plotted on the side B. At the end of plotting, the side A is made the plotting surface and the side B the display surface.



LEGAL STATUS

[Date of request for examination]
 [Date of sending the examiner's decision of rejection]
 [Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]
 [Date of final disposal for application]
 [Patent number]
 [Date of registration]
 [Number of appeal against examiner's decision of rejection]
 [Date of requesting appeal against examiner's decision of rejection]
 [Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

(43)公開日 平成6年(1994)1月28日

(51)Int.Cl.⁵
G 0 6 F 3/153

識別記号 336 B 庁内整理番号 7165-5B

FI

技術表示箇所

審査請求 未請求 請求項の数 1 (全 12 頁)

(21)出願番号 特願平4-172650

(22)出願日 平成4年(1992)6月30日

(71)出願人 000005223
富士通株式会社
神奈川県川崎市中原区上小田中1015番地

(72)発明者 小林 智博
神奈川県横浜市港北区新横浜2丁目4番19
号 株式会社富士通プログラム技研内

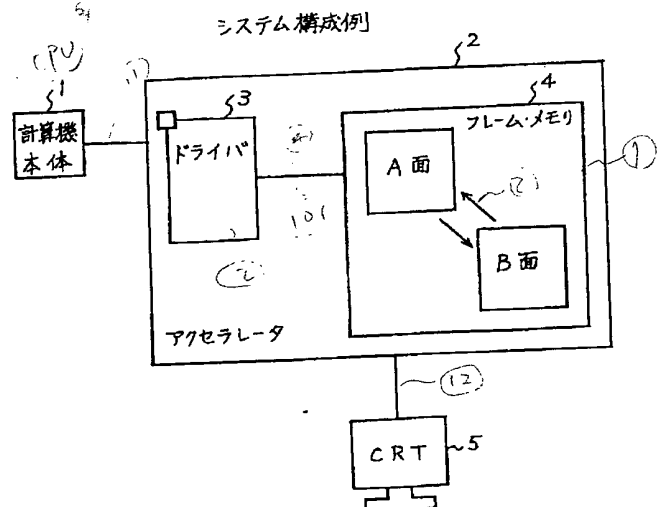
(74)代理人 弁理士 京谷 四郎

(54)【発明の名称】 グラフィックス・システム

(57) 【要約】

【目的】 アプリケーション・プログラムにおいてダブル・バッファリング機能を意識した複雑な制御を不要にすると共に、アプリケーション・プログラムから出力された表示内容が常にユーザ画面上に表示されるようにすること。

【構成】 A面が表示面、B面が描画面とすると、A面の内容がディスプレイ5に表示される。ドライバ3は、計算機本体1から描画要求が送られて来ると、表示面（この場合はA面）の全内容を描画面（この場合はB面）にコピーし、コピー終了後に描画要求で指定された画像をB面に描画し、描画終了後にA面を描画面にし、B面を表示面にする。



【特許請求の範囲】

【請求項1】 A面メモリおよびB面メモリを有するフレーム・メモリ(4)並びにフレーム・メモリ(4)に対して画像を書き込むドライバ処理手段(3)を備えるアクセラレータ(2)と、
アクセラレータ(2)に対して描画要求を送る計算機本体(1)と、
フレーム・メモリ(4)の中の表示面とされているメモリの内容を表示するディスプレイ(5)とを具備するグラフィックス・システムであって、
ドライバ処理手段(3)は、描画要求が送られた来た時、A面メモリとB面メモリの内の何れが表示面とされているかを調べ、
表示面とされているメモリの全内容を、描画面とされている他方のメモリにコピーし、
コピー終了後に、描画面とされているメモリに当該描画要求で指定されている画像を書き込み、
書き込み終了後に、描画面となっているメモリを表示面に、表示面になっているメモリを描画面に切り換える処理を行うように構成されていることを特徴とするグラフィックス・システム。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、グラフィックス・システムにおけるダブル・バッファリング制御に関するものである。近年の3次元グラフィックスにおけるアプリケーションでは、ダブル・バッファリング機能を使用することが多くなってきている。このため、ユーザが容易に使用できるダブル・バッファリング機能が必要とされる。

【0002】

【従来の技術】図8は従来技術の機能概略を説明する図である。同図において、1は計算機本体、3はドライバ、4はフレーム・メモリ、5はCRTを示している。アプリ(アプリケーションの略)1, 2, 3は計算機本体1によって実行される。ドライバ3は、画像をフレーム・メモリ4に書き込むものである。フレーム・メモリ4は2個のメモリを有しており、一方のメモリはA面と呼ばれ、他方のメモリはB面と呼ばれる。A面とB面のうちの一方が表示面になり、他方が描画面となる。表示面に書かれている画像がCRT5に表示される。

【0003】アプリ1がA面に描画と言う要求をドライバ3に送ると、ドライバ3は要求で指定された画像をA面に書き込む。アプリ1がA面を表示と言う要求をドライバ3に送ると、ドライバ3はA面をCRT5に表示する。アプリ2がB面に描画と言う要求をドライバ3に送ると、ドライバ3は要求で指定された画像をB面に書き込む。アプリ2がB面を表示と言う要求をドライバ3に送ると、ドライバ3はB面をCRT5に表示する。アプリ3がA面に描画と言う要求をドライバ3に送ると、ド

ライバ3は要求で指定された画像をA面に書き込む。アプリ3がA面を表示と言う要求をドライバ3に送ると、ドライバ3はA面をCRT5に表示する。

【0004】図9、図10は従来方式の動作例を説明する図である。アプリ①がB面に描画と言う要求をドライバに送ると、A面が表示面、B面が描画面となり、B面にアプリ①からの画像が書き込まれる。この状態は図9(a)に示される。この状態では、A面には何も書き込まれていないので、ユーザが見る表示画面には何も表示されない。なお、①が付けられている四角形は、アプリ①に割り当てられた領域を示している。(a)の状態、アプリ①がB面を表示と言う要求をドライバに送ると、A面が描画面、B面が表示面となる。この状態は図9(b)に示される。この状態では、B面に書かれたアプリ①の画像が表示される。

【0005】(b)の状態、アプリ②がA面に描画と言う要求をドライバに送ると、A面が描画面、B面が表示面となり、A面にアプリ②からの画像が書き込まれる。この状態は図10(c)に示される。この状態では、B面に書かれたアプリ①の画像が表示される。(c)の状態、アプリ②がA面を表示と言う要求をドライバに送ると、A面が表示面、B面が描画面となる。この状態は図10(d)に示される。この状態では、A面に書かれたアプリ②の画像が表示される。図10(d)の状態では、ユーザが見る表示画面には、アプリ①の表示内容が保証されない。

【0006】

【発明が解決しようとする課題】従来のドライバのダブル・バッファリング制御では、ユーザ側が常に描画面と表示面を意識して画面の切換制御を行っていたので、アプリケーション・プログラムが複雑化してしまった。また、同時に複数のプログラムを起動する場合やウィンドウを移動させる場合、マウスからイベントが上がった場合には、アプリケーション・プログラム側では制御しきれなくなり、画面上の表示内容が保証されなかった。本発明は、この点に鑑みて創作されたものであって、アプリケーション・プログラムにおいてダブル・バッファリング機能を意識した複雑な制御を行うことを不要にすると共に、アプリから出力された表示内容がユーザ画面上に常に表示されるようにすることを目的としている。

【0007】

【課題を解決するための手段】請求項1のグラフィックス・システムは、A面メモリおよびB面メモリを有するフレーム・メモリ(4)並びにフレーム・メモリ(4)に対して画像を書き込むドライバ処理手段(3)を備えるアクセラレータ(2)と、アクセラレータ(2)に対して描画要求を送る計算機本体(1)と、フレーム・メモリ(4)の中の表示面とされているメモリの内容を表示するディスプレイ(5)とを具備するグラフィックス・システムであって、ドライバ処理手段(3)は、描画要求が送られた来た

時、A面メモリとB面メモリの内の何れが表示面とされているかを調べ、表示面とされているメモリの全内容を、描画面とされている他方のメモリにコピーし、コピー終了後に、描画面とされているメモリに当該描画要求で指定されている画像を書き込み、書き込み終了後に、描画面となっているメモリを表示面に、表示面になっているメモリを描画面に切り換える処理を行うように構成されていることを特徴とするものである。

【0008】

【作用】A面に家の絵が書かれており、B面には何も書かれていないと仮定する。また、A面が表示面、B面が描画面であると仮定する。このような状態の下では、A面の家の絵がディスプレイ(5)の画面に表示される。

【0009】計算機本体(1)から松の木の絵を書きたいと言う描画要求が送られて来ると、ドライバ処理手段(3)は、表示面とされているA面の全内容を描画面とされているB面にコピーし、コピー終了後に松の木の絵をB面に描画し、A面を描画面、B面を表示面に切り換える。そうすると、ディスプレイ(5)の画面には家の絵と松の木の絵が表示される。

【0010】

【実施例】図1は本発明のシステム構成を示すものである。同図において、1は計算機本体、2はアクセラレータ、3はドライバ、4はフレーム・メモリ、5はCRTをそれぞれ示している。アクセラレータ2は高速描画機構であり、内部に高速のプロセッサを有している。ドライバ3の主体はドライバ用プログラムであり、このプログラムはプロセッサによって実行される。ドライバ用プログラムは、計算機本体1からアクセラレータ2にダウンロードされる。

【0011】フレーム・メモリ4はA面とB面とを有している。A面とB面のうちの一方が表示面となり、他方が描画面となる。表示面に格納されている画像はCRT5に表示される。ドライバ3は、A面とB面の内の何れが表示面かを調べたり、表示面の画像を描画面にコピーしたり、描画面に対して描画を行ったり、描画面と表示面の切換えを行ったりするための処理を行う。

【0012】図2は本発明の機能概略を説明する図である。同図において、1は計算機本体、3はドライバ、4はフレーム・メモリ、5はCRTを示している。アプリ1、2、3は計算機本体1で実行される。初期状態においては、A面が描画面、B面が表示面になっていると仮定する。アプリ1が描画要求をドライバ3に送ると、ドライバ3はB面の内容をA面にコピーし、A面にアプリ1からの画像を書き込み、A面を表示面、B面を描画面とする。この結果、A面に書き込まれているアプリ1からの画像がCRT5に表示される。

【0013】この状態の下で、アプリ2が描画要求をドライバ3に送ると、ドライバ3はA面の内容をB面にコピーし、B面にアプリ2からの画像を書き込み、A面を

描画面、B面を表示面とする。この結果、B面に書き込まれているアプリ1、2からの画像がCRT5に表示される。

【0014】この状態の下で、アプリ3が描画要求をドライバ3に送ると、ドライバ3はB面の内容をA面にコピーし、A面にアプリ3からの画像を書き込み、A面を表示面、B面を描画面とする。この結果、A面に書き込まれているアプリ1、2、3からの画像がCRT5に表示される。

10 【0015】図3は本発明のドライバの処理フローを説明する図である。なお、図3はドライバの処理の一部を示している。ステップS1では、ダブル・バッファ・モードが付いているか否かを調べる。Yesの場合はステップS2に進み、Noの場合はステップS10に進む。Noの場合はシングル・バッファリング機能となる。ステップS2では、アプリがダブル・バッファリング機能を指定しているか否かを調べる。Yesの場合はステップS3に進み、Noの場合はステップS10に進む。

20 【0016】ステップS3では、現在の表示面がA面か否かを調べる。Yesの場合はステップS4に進み、Noの場合にはステップS6に進む。なお、A面とB面のうちの何れが表示面になっているかは、後述のフラグ領域に記録されている。ステップS4では、B面を描画面に指定する。ステップS5では、A面からB面に全画面コピーを行う。ステップS6では、A面を描画面に設定する。ステップS7では、B面からA面に全画面コピーを行う。

30 【0017】ステップS8では、描画面に対して描画を行う。ステップS9では、描画面を表示面に切り換える。ステップS10では、描画面に対して描画を行う。

【0018】図4、図5は本発明の動作例を説明する図である。図4(a)においては、A面にはアプリ①の画像が書き込まれており、A面が表示面、B面が描画面となっている。ユーザが見る表示画面には、A面に書き込まれているアプリ①の画像が表示される。

【0019】この状態の下で、アプリ②から描画要求があると、それまでの表示内容を保証するために、A面の全内容がB面にコピーされる。この状態は、図4(b)に示される。次に、描画面とされているB面に、アプリ②からの画像を描画する。この状態は、図4(c)に示される。描画終了後、A面を描画面に、B面を表示面とする。この結果、ユーザが見る表示画面には、B面に書き込まれているアプリ①、②からの画像が表示される。この状態は、図5(d)に示される。

40 【0020】図5(e)はドライバの持つフラグ領域を示す図である。ドライバはフラグ領域を参照しながら処理を行う。フラグ領域は32ビットの大きさを有しており、ビット0はアプリ表示フラグ、ビット1は表示面フラグ、ビット2は描画面フラグである。1のアプリ表示フラグは、現在1つ以上のアプリが表示されていること

を示す。0の表示面フラグはA面が表示面であることを示し、1の表示面フラグはB面が表示面であることを示す。0の描画面フラグはA面が描画面であることを示し、1の描画面フラグはB面が描画面であることを示す。図5(e)のフラグ領域の状態は、図5(d)の状態に対応している。

【0021】図6、図7は本発明の他の動作例を説明する図である。図6(a)においては、A面にはアプリ①の画像が書き込まれており、B面にはアプリ①、②の画像が書き込まれており、A面が描画面、B面が表示面とな

っている。ユーザが見る表示画面には、B面に書き込まれているアプリ①、②の画像が表示される。
【0022】この状態の下で、アプリ①から描画要求があると、B面の全内容がA面にコピーされる。この状態は、図6(b)に示される。次に、描画面とされているA面に、アプリ①からの画像を描画する。この状態は、図6(c)に示される。描画終了後、A面を表示面に、B面を描画面とする。この結果、ユーザが見る表示画面には、A面に書き込まれているアプリ①、②からの画像が表示される。この状態は、図7(d)に示される。図7(e)のフラグ領域の状態は、図7(d)に対応している。即ち、表示面フラグ=0(A面が表示面)、描画面フラグ=1(B面が描画面)になっている。

【0023】

【発明の効果】以上の説明から明らかなように、本発明によれば、

* a) アプリケーション・プログラムに複雑な処理を加えることなく、ユーザが容易にダブル・バッファリング機能を使用できる。

b) 複数のプログラムを同時にダブル・バッファリング機能を使用して起動することが出来る。

と言う顕著な効果を得ることが出来る。

【図面の簡単な説明】

【図1】本発明のシステム構成例を示す図である。

【図2】本発明の機能概略を説明する図である。

【図3】本発明のドライバの処理フローを示す図である。

【図4】本発明の動作例を説明する図である。

【図5】本発明の動作例(続き)を説明する図である。

【図6】本発明の他の動作例を説明する図である。

【図7】本発明の他の動作例(続き)を説明する図である。

【図8】従来技術の機能概略を説明する図である。

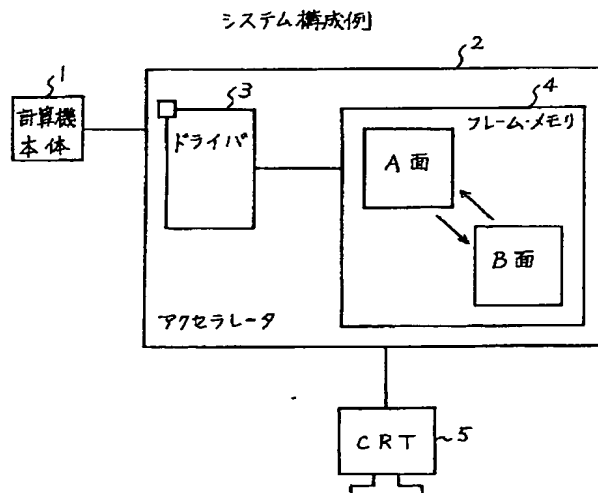
【図9】従来方式の動作例を説明する図である。

【図10】従来方式の動作例(続き)を説明する図である。

【符号の説明】

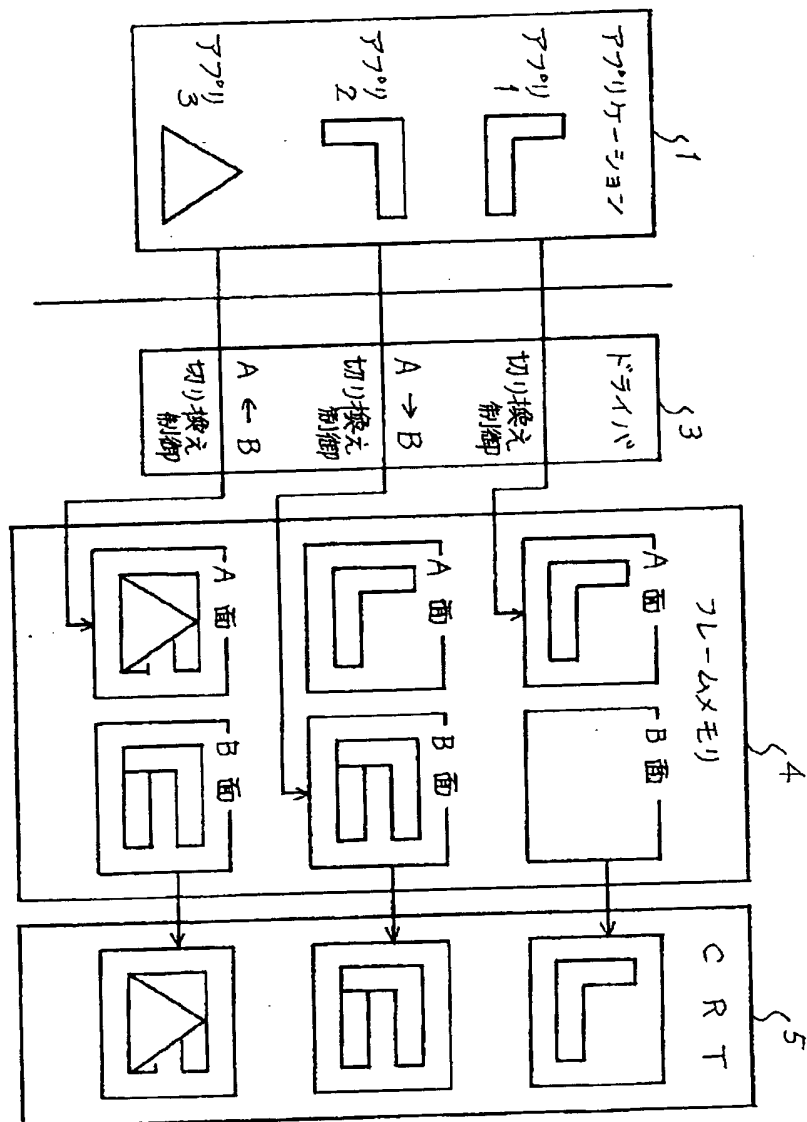
- 1 計算機本体
- 2 アクセラレータ
- 3 ドライバ
- 4 フレーム・メモリ
- 5 CRT

【図1】



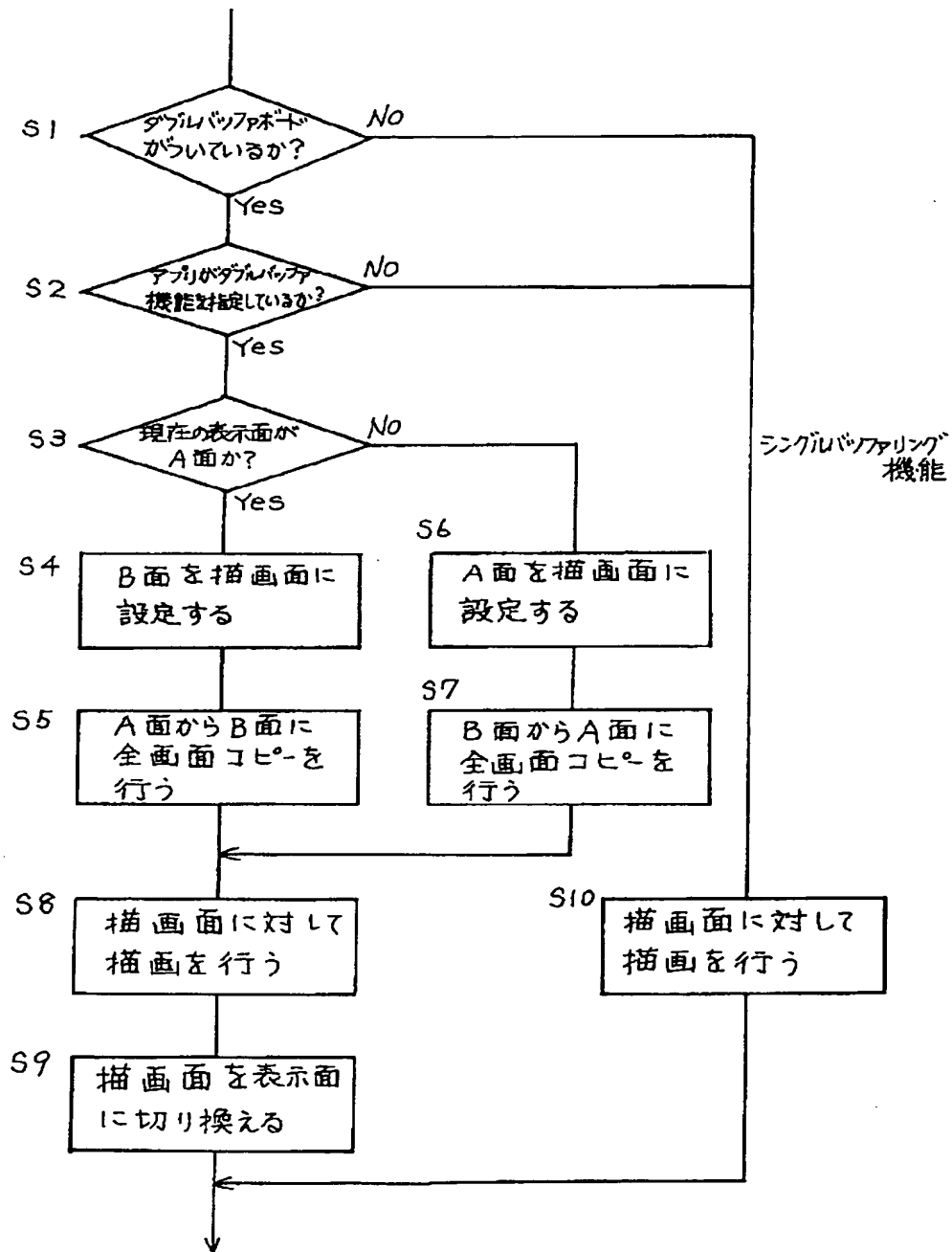
【図2】

本発明の機能概略



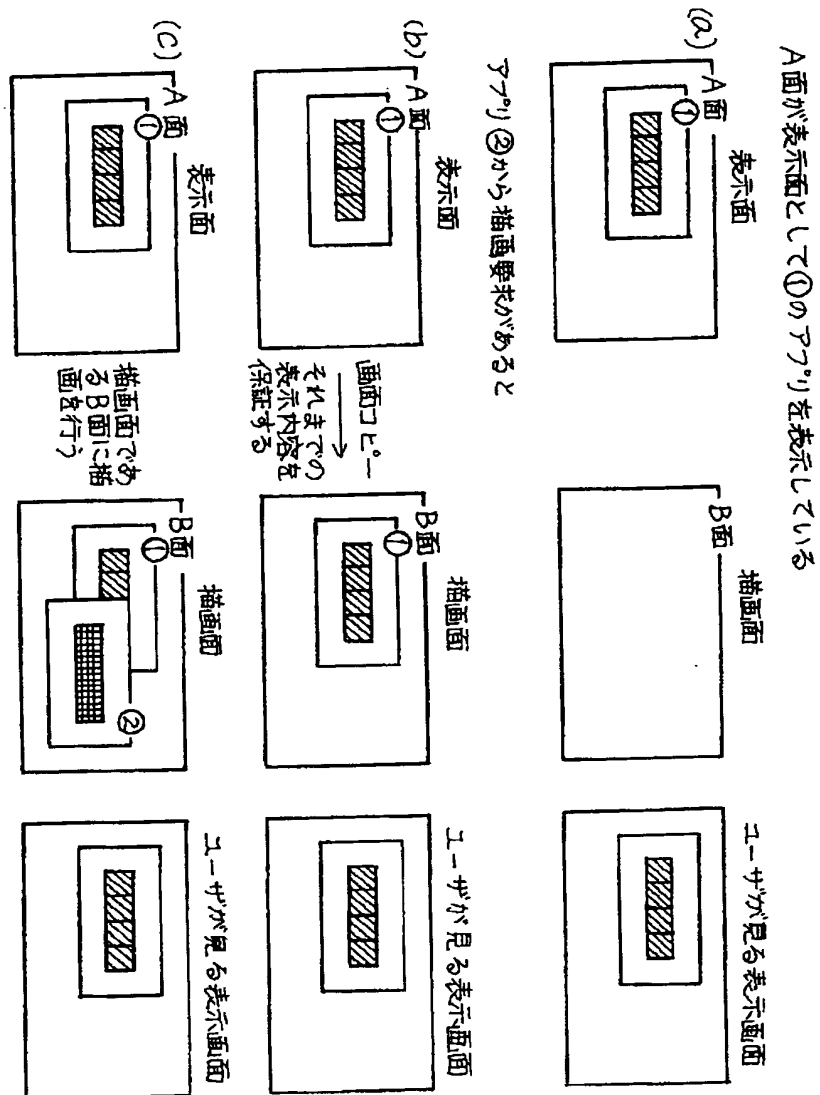
【図3】

ドライバの処理フロー



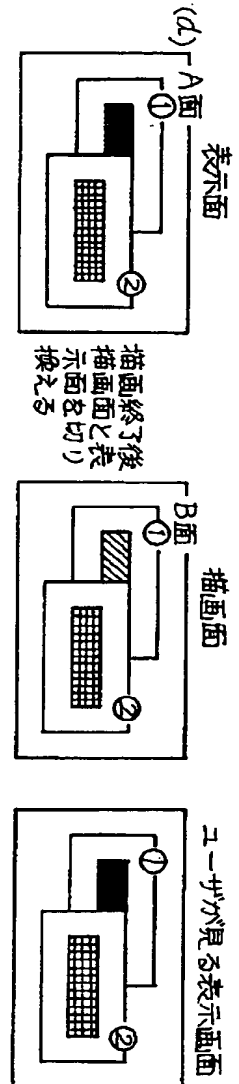
【図4】

本発明の動作例



【図7】

本発明の他の動作例(続き)



(e)

3	1	0	2	1	0
...
0	1	0	1	1	1

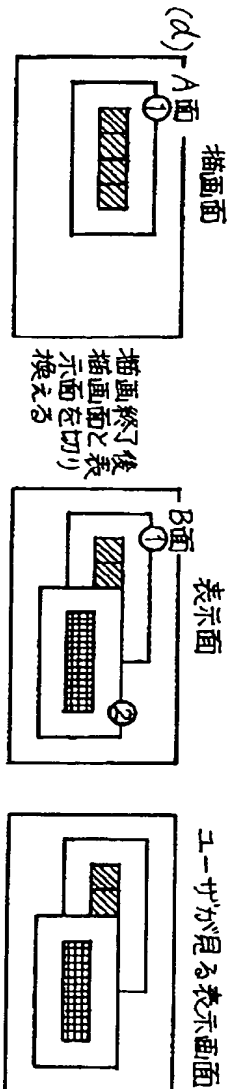
0 : A面
1 : B面

描画面フラグ
表示面フラグ
アプリ表示フラグ

A面が表示面・B面が描画面であることを識別するフラグをたてる

【図5】

本発明の動作例(続き)



(e)

3	1	0	2	1	0
...
0	1	0	1	1	1

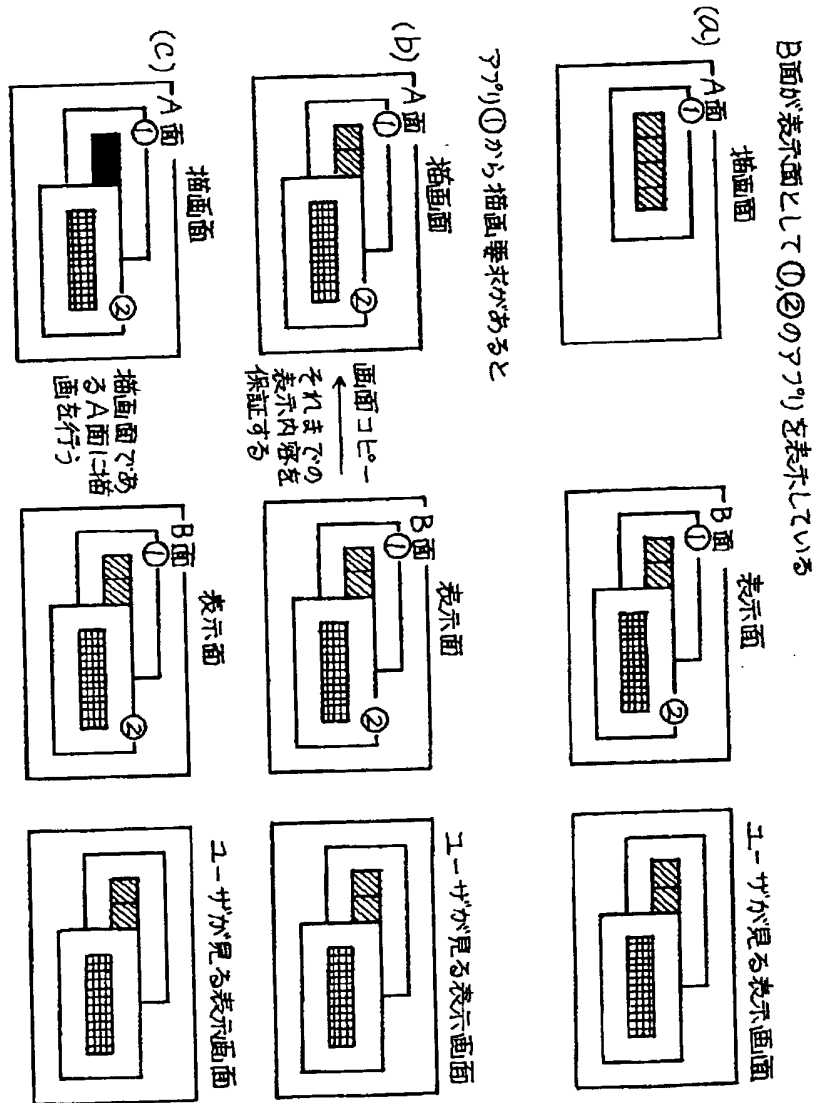
0 : A面
1 : B面

描画面フラグ
表示面フラグ
アプリ表示フラグ

A面が描画面・B面が表示面であることを識別するフラグをたてる

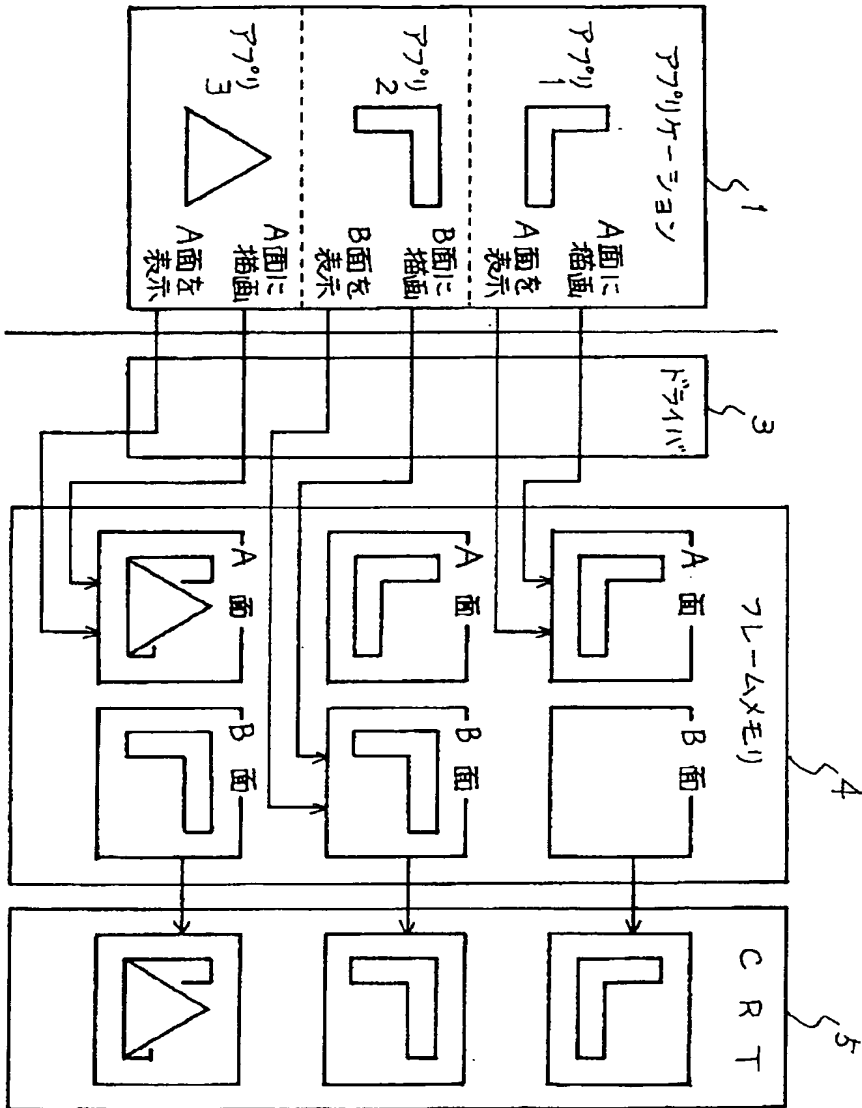
【図6】

本発明の他の動作例



【図8】

従来技術の機能概略

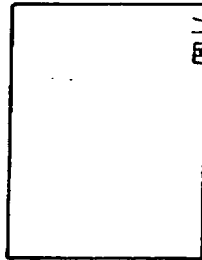


【図9】

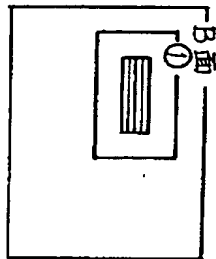
従来方式の動作例

アプリ①が描画面をB面に指定し描画する

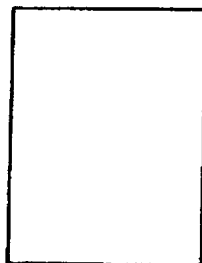
(a) A面 表示面



描画面

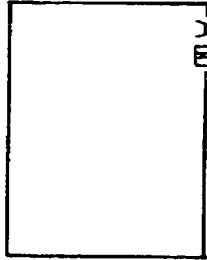


ユーザが見る表示画面

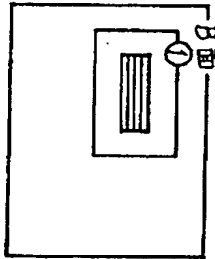


アプリ①が描画面と表示面を切り換える

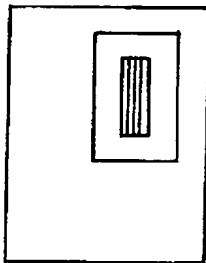
(b) A面 描画面



表示面



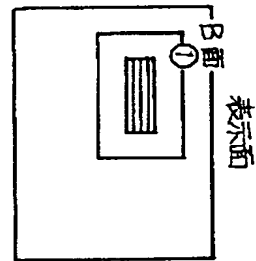
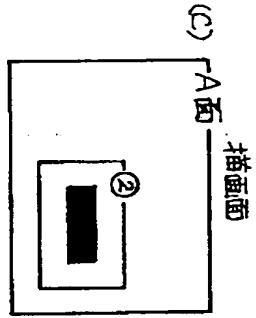
ユーザが見る表示画面



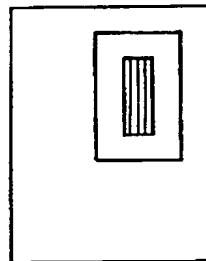
【図10】

従来方式の動作例(続き)

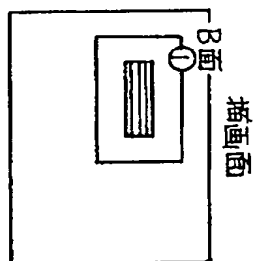
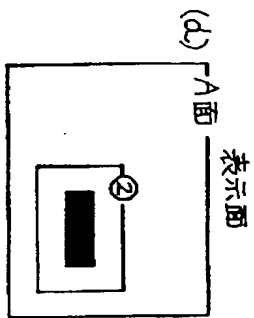
その状態でアプリ②が描画面にA面を指定する



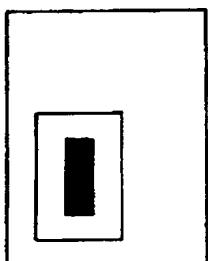
ユーザが見る表示画面



アプリ②が描画面と表示面を切り換えると



ユーザが見る表示画面



アプリ①の表示内容
が保証されない